

12. Who is the real author of Hamlet?

	Doc	Words	Author
Training	1	W1 W2 W3 W4 W5	C (Christopher Marlowe)
	2	W1 W1 W4 W3	C (Christopher Marlowe)
	3	W1 W2 W5	C (Christopher Marlowe)
	4	W5 W6 W1 W2 W3	W (William Stanley)
	5	W4 W5 W6	W (William Stanley)
	6	W4 W6 W3	F (Francis Bacon)
	7	W2 W2 W4 W3 W5 W5	F (Francis Bacon)
Test	8 (Hamlet)	W1 W4 W6 W5 W3	?

Training

Priors:

$P(X)$ = The probability of an Author X

= Number of Author X / total number of Authors = N_x / N

Note:

$P(C)$ = The probability of Author C = 3/7 (i.e., 3 C- Authors / total Authors)

$P(W)$ = The probability of Author W = 2/7 (i.e., 2 W- Authors / total Authors)

$P(F)$ = The probability of Author F = 2/7 (i.e., 2 F- Authors / total Authors)

Conditional probabilities:

$P(w|x)$ = If a document belongs to Author x, the probability that the document has word w.

= The probability the word w appears on the Author x document.

= $(\text{count}(w, x) + 1) / (\text{count}(x) + |V|)$

Note:

Original definition of **$P(w|x) = \text{count}(w, x) / \text{count}(x)$**

$\text{count}(w, x)$: how many times the word w appears on the x Author documents.

$\text{count}(x)$: how many words on the x Author documents.

|V|: number of vocabularies = number of different words

Tunable knobs (i.e., parameters) of Naive Bayes

1 and |V| are used for Laplace Smoothing to prevent the possibility of letting $P(w|x)$ have value of 0 or 1.

Other values can be used to replace 1 and |V|.

The Test only has 5 words: **W1, W4, W6, W5, W3.**

- $P(W1|C) = (\text{count}(W1, C) + 1) / (\text{count}(C) + |V|) = (4+1) / (12+6) = 5/18$

= The probability the word "W1" appear on the Author "C" documents.

Note:

4: how many times the word "W1" appear on the 3 C- Authors.

12: how many words in the 3 C- Authors.

6: is number of vocabularies: **W1, W2, W3, W4, W5, W6**

- $P(W1|W) = (\text{count}(W1, W) + 1) / (\text{count}(W) + |V|) = (1+1) / (8+6) = 2/14$

= The probability the word "W1" appear on the Author "W" documents.

Note:

1: how many times the word "W1" appear on the 2 W- Authors.

8: how many words in the 2 W- Authors.

6: is number of vocabularies: **W1, W2, W3, W4, W5, W6**

- $P(W1|F) = (\text{count}(w1, F) + 1) / (\text{count}(F) + |V|) = (0+1)/(9+6) = 1/15$

= The probability the word "W1" appear on the Author "F" documents.

Note:

0: how many times the word "W1" appear on the 2 F- Authors.

9: how many words in the 2 F- Authors.

6: is number of vocabularies: **W1, W2, W3, W4, W5, W6**

- $P(W3|C) = (\text{count}(W3, C) + 1) / (\text{count}(C) + |V|) = (2+1)/(12+6) = 3/18$

- $P(W3|W) = (\text{count}(W3, W) + 1) / (\text{count}(W) + |V|) = (1+1)/(8+6) = 2/14$
- $P(W3|F) = (\text{count}(W3, F) + 1) / (\text{count}(F) + |V|) = (2+1) / (9+6) = 3/15$
- $P(W4|C) = (\text{count}(W4, C) + 1) / (\text{count}(C) + |V|) = (2+1) / (12+6) = 3/18$
- $P(W4|W) = (\text{count}(W4, W) + 1) / (\text{count}(W) + |V|) = (1+1)/(8+6) = 2/14$
- $P(W4|F) = (\text{count}(W4, F) + 1) / (\text{count}(F) + |V|) = (2+1) / (9+6) = 3/15$
- $P(W5|C) = (\text{count}(W5, C) + 1) / (\text{count}(C) + |V|) = (2+1) / (12+6) = 3/18$
- $P(W5|W) = (\text{count}(W5, W) + 1) / (\text{count}(W) + |V|) = (2+1)/(8+6) = 3/14$
- $P(W5|F) = (\text{count}(W5, F) + 1) / (\text{count}(F) + |V|) = (2+1) / (9+6) = 3/15$
- $P(W6|C) = (\text{count}(W6, C) + 1) / (\text{count}(C) + |V|) = (0+1)/(12+6) = 1/18$
- $P(W6|W) = (\text{count}(W6, W) + 1) / (\text{count}(W) + |V|) = (2+1)/(8+6) = 3/14$
- $P(W6|F) = (\text{count}(W6, F) + 1) / (\text{count}(F) + |V|) = (1+1)/(9+6) = 2/15$

Test

Decide whether **d8** (i.e., **document 8**) belongs to **Author C** or **Author W** or **Author F**.

• Step 1: Analysis

A. The probability of **d8** (i.e., **document 8**) belonging to **Author C**

$$P(C|d8) = P(C) * P(d8|C) / P(d8)$$

=> Applying Bayes Theorem

$$= P(C) * P(W1 \cap W4 \cap W6 \cap W5 \cap W3 | C) / P(d8)$$

=> Applying Naive Bayes Theorem

$$\propto (P(C) * P(W1|C) * P(W4|C) * P(W6|C) * P(W5|C) * P(W3|C)) / P(d8)$$

$$= P(C) * P(W1|C) * P(W4|C) * P(W6|C) * P(W5|C) * P(W3|C) / P(d8)$$

=> Applying Compare Model

$$P(C|d8) \propto P(C) * P(W1|C) * P(W4|C) * P(W6|C) * P(W5|C) * P(W3|C)$$

$$= 3/7 * 5/18 * 3/18 * 1/18 * 3/18 * 3/18 = \mathbf{0.00003061924}$$

B. The probability of **document 8** belonging to **Author W**.

=> Applying Naive Bayes Theorem

$$P(W|d8) \propto (P(W) * P(W1|W) * P(W4|W) * P(W6|W) * P(W5|W) * P(W3|W)) / P(d8)$$

$$= P(W) * P(W1|W) * P(W4|W) * P(W6|W) * P(W5|W) * P(W3|W) / P(d8)$$

==> Applying Compare Model

$$P(W|d8) \propto P(W) * P(W1|W) * P(W4|W) * P(W6|W) * P(W5|W) * P(W3|W)$$

$$= 2/7 * 2/14 * 2/14 * 3/14 * 3/14 * 2/14 = \mathbf{0.00003824936}$$

C. The probability of document 8 belonging to Author F.

==> Applying Naive Bayes Theorem

$$P(F|d8) \propto (P(F) * P(W1|F) * P(W4|F) * P(W6|F) * P(W5|F) * P(W3|F)) / P(d8)$$

$$= P(F) * P(W1|F) * P(W4|F) * P(W6|F) * P(W5|F) * P(W3|F) / P(d8)$$

==> Applying Compare Model

$$P(F|d8) \propto P(F) * P(W1|F) * P(W4|F) * P(W6|F) * P(W5|F) * P(W3|F)$$

$$= 2/7 * 1/15 * 3/15 * 2/15 * 3/15 * 3/15 = \mathbf{0.00002031746}$$

Step 2: Conclusion

Document 8(Hamlet) should belong to the Author W.

```

Text_Classifier.ipynb
File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text
[+] Pipeline(memory=None,
  steps=[('cleaner', <_main__predictors object at 0x7fa155788f90>),
        ('vectorizer',
         CountVecorizer(analyzer='word', binary=False,
                        decode_error='strict',
                        dtype=<class 'numpy.int64'>, encoding='utf-8',
                        input='content', lowercase=True, max_df=1.0,
                        max_features=None, min_df=1,
                        ngram_range=(1, 1), preprocessor=None,
                        stop_words=None, strip_accents=None,
                        token_pattern=r't...\b\\w+\\b',
                        tokenizer=<function spacy_tokenizer at 0x7fa152585050>,
                        vocabulary=None)),
        ('classifier',
         LogisticRegression(C=1.0, class_weight=None, dual=False,
                           fit_intercept=True, intercept_scaling=1,
                           l1_ratio=None, max_iter=100,
                           multi_class='auto', n_jobs=None,
                           penalty='l2', random_state=None,
                           solver='lbfgs', tol=0.0001, verbose=0,
                           warm_start=False))],
  verbose=False)

[349] New_Value = ["w1 w4 w6 w5 w3"]
      predicted1 = pipe.predict(New_Value) #New data
      print(predicted1)

['W']
  
```

0s completed at 12:55 AM

Type here to search 60°F Clear 12:56 AM 6/24/2021